

WHAT IS CLAIMED IS:

1 1. A terminal device, comprising:
2 a device driver section for controlling an interface section
3 that is connected to a network;
4 a protocol stack section that is connected to the network
5 via the device driver section based on a communication protocol
6 at higher than or equal to layer3 of OSI; and
7 a middleware section that is interposed between the protocol
8 stack section and the device driver section;
9 wherein the middleware section includes a sending section
10 that, if a send request for a frame to be sent (hereinafter, referred
11 to as "send frame") is issued from the protocol stack section,
12 determines a send priority of the send request based on header
13 information at layer2 to layer4 of OSI within the send frame, and
14 then outputs the send request to the device driver section according
15 to the send priority determined.

1 2. A terminal device according to claim 1, wherein the
2 middleware section further includes a receiving section that, if
3 a receive request for a frame to be received (hereinafter, referred
4 to as "receive frame") is issued from the device driver section,
5 determines a receive priority of the receive request based on header
6 information at layer2 to layer4 of OSI within the receive frame,
7 and then outputs the receive request to the protocol stack section
8 according to the receive priority determined.

1 3. A terminal device according to claim 1, further
2 comprising:
3 a cache table in which specific session information is
4 registered in advance; and

5 multiple FIFOs each corresponding to the send priority,
6 wherein the sending section includes:
7 a header comparison section that, if a send request
8 for a send frame is issued from the protocol stack section,
9 determines a send priority of the send request by searching the
10 cache table based on header information at layer2 to layer4 of
11 OSI within the send frame, and then queues the send request to
12 one of the multiple FIFOs corresponding to the send priority
13 determined; and
14 a synthesis section that synthesizes send requests
15 outputted from the multiple FIFOs according to the send priority
16 of the one of the multiple FIFOs to which the send request is queued,
17 and then outputs a synthesized send request to the device driver
18 section.

1 4. A terminal device according to claim 2, further
2 comprising:
3 a cache table in which specific session information is
4 registered in advance; and
5 multiple FIFOs each corresponding to the receive priority,
6 wherein the receiving section includes:
7 a header comparison section that, if a receive request
8 for a receive frame is issued from the device driver section,
9 determines a receive priority of the receive request by searching
10 the cache table based on header information at layer2 to layer4
11 of OSI within the receive frame, and then queues the receive request
12 to one of the multiple FIFOs corresponding to the receive priority
13 determined; and
14 a synthesis section that synthesizes receive requests
15 outputted from the multiple FIFOs according to the receive priority

16 of the one of the multiple FIFOs to which the receive request is
17 queued, and then outputs a synthesized receive request to the
18 protocol stack section.

1 5. A terminal device, comprising:
2 a device driver section for controlling an interface section
3 that is connected to a network;
4 a protocol stack section that is connected to the network
5 via the device driver section based on a communication protocol
6 at higher than or equal to layer3 of OSI; and
7 a middleware section that is interposed between the protocol
8 stack section and the device driver section;
9 wherein the middleware section includes a sending section
10 that:
11 if a send request to a predetermined destination for a
12 specific packet defined by a communication protocol at higher than
13 or equal to layer5 of OSI is issued from the protocol stack section
14 in advance and if the send request is a first one of consecutive
15 send requests, checks on header information of the specific packet,
16 then registers into the cache table session information extracted
17 from headers at layer2 to layer4 of OSI within a send frame carrying
18 the specific packet, raises a send priority of the send request,
19 and outputs the send request to the device driver section; and
20 if the send request is among the consecutive send requests
21 other than the first one and if session information extracted from
22 headers at layer2 to layer4 of OSI within a send frame carrying
23 the specific packet is registered in the cache table, raises a
24 send priority of the send request, and outputs the send request
25 to the device driver section.

1 6. A terminal device according to claim 5, wherein the
2 middleware section further includes a receiving section that:

3 if a receive request from a predetermined source for a
4 specific packet defined by a communication protocol at higher than
5 or equal to layer5 of OSI is issued from the device driver section
6 in advance and if the receive request is a first one of consecutive
7 receive requests, checks on header information of the specific
8 packet, then registers into the cache table session information
9 extracted from headers at layer2 to layer4 of OSI within a receive
10 frame carrying the specific packet, raises a receive priority of
11 the receive request, and outputs the receive request to the protocol
12 stack section; and

13 if the receive request is among the consecutive receive
14 requests other than the first one and if session information
15 extracted from headers at layer2 to layer4 of OSI within a receive
16 frame carrying the specific packet is registered in the cache table,
17 raises a receive priority of the receive request, and outputs the
18 receive request to the protocol stack section.

1 7. A terminal device according to claim 5, wherein the
2 middleware section further includes a monitor section that:

3 if the session information is registered into the cache table,
4 monitors the cache table; and

5 if the send request for a send frame carrying the session
6 information is not issued from the protocol stack section within
7 a predetermined time, delete the session information within the
8 cache table.

1 8. A terminal device according to claim 6, wherein the

2 middleware section further includes a monitor section that:
3 if the session information is registered into the cache table,
4 monitors the cache table; and
5 if the receive request for a receive frame carrying the
6 session information is not issued from the device driver section
7 within a predetermined time, delete the session information within
8 the cache table.

1 9. A terminal device, comprising:
2 a device driver section for controlling an interface section
3 that is connected to a network;
4 a protocol stack section that is connected to the network
5 via the device driver section based on a communication protocol
6 at higher than or equal to layer3 of OSI;
7 a middleware section that is interposed between the protocol
8 stack section and the device driver section;
9 a first cache table in which a first session information
10 is previously registered;
11 a second cache table that is used upon establishment of a
12 session;
13 a first FIFO section for storing high-priority send data
14 in a FIFO format; and
15 a second FIFO section for storing low-priority send data
16 in a FIFO format,
17 wherein the middleware section includes:
18 a first checking means for, if a send request for a
19 send frame is issued from the protocol stack section, checking
20 whether a second session information that is extracted from headers
21 at layer2 to layer4 of OSI within the send frame is registered
22 in the second cache table;

23 a first queueing means for, if the first checking means
24 determines that the second session information is registered,
25 queueing the send request to the first FIFO section;

26 a second checking means for, if the first checking
27 means determines that the second session information is not
28 registered, checking whether the second session information is
29 registered in the first cache table;

30 a third checking means for, if the second checking
31 means determines that the second session information is registered,
32 checking whether the send frame includes a predetermined specific
33 packet at higher than or equal to layer 5 of OSI;

34 a second queueing means for, if the third checking
35 means determines that the send frame includes the predetermined
36 specific packet, judging that a high-priority session is
37 established, registering the second session information into the
38 second cache table, and queueing the send request to the first
39 FIFO section;

40 a third queueing means for, if the second checking
41 means determines that the second session information is not
42 registered, queueing the send request to the second FIFO section;
43 and

44 send-requesting means for outputting to the device
45 driver section the send request queued to the first FIFO section
46 prior to the send request queued to the second FIFO section.

1 10. A terminal device according to claim 9, comprising:
2 a third FIFO section for storing high-priority receive data
3 in a FIFO format; and
4 a fourth FIFO section for storing low-priority receive data
5 in a FIFO format,

6 wherein the middleware section includes:

7 a fourth checking means for, if a receive request for
8 a receive frame is issued from the device driver section, checking
9 whether a third session information that is extracted from headers
10 at layer2 to layer4 of OSI within the receive frame is registered
11 in the second cache table;

12 a fourth queueing means for, if the fourth checking
13 means determines that the third session information is registered,
14 queueing the receive request to the third FIFO section;

15 a fifth checking means for, if the fourth checking
16 means determines that the third session information is not
17 registered, checking whether the third session information is
18 registered in the first cache table;

19 a sixth checking means for, if the fifth checking means
20 determines that the third session information is registered,
21 checking whether the receive frame includes a specific packet;

22 a fifth queueing means for, if the sixth checking means
23 determines that the receive frame includes the specific packet,
24 judging that a high-priority session is established, registering
25 the third session information into the second cache table, and
26 queueing the receive request to the third FIFO section;

27 a sixth queueing means for, if the fifth checking means
28 determines that the third session information is not registered,
29 queueing the receive request to the fourth FIFO section; and

30 receive-requesting means for outputting to the
31 protocol stack section the receive request queued to the third
32 FIFO section prior to the receive request queued to the fourth
33 FIFO section.

1 11. A terminal device according to claim 9, wherein the

2 middleware section further includes monitor means for:
3 if the second session information is registered into the
4 second cache table, monitoring the second cache table; and
5 if the send request for a send frame carrying the second
6 session information is not issued from the protocol stack section
7 within a predetermined time, deleting the second session
8 information within the second cache table.

1 12. A terminal device according to claim 10, wherein the
2 middleware section further includes monitor means for:
3 if the third session information is registered into the
4 second cache table, monitoring the second cache table; and
5 if the receive request for a receive frame carrying the third
6 session information is not issued from the device driver section
7 within a predetermined time, deleting the third session information
8 within the second cache table.

1 13. A terminal device according to claim 1, wherein the
2 middleware section further includes means for, if the send request
3 is sent to the device driver section, sending the send request
4 via a program interface with respect to the protocol stack section.

1 14. A terminal device according to claim 2, wherein the
2 middleware section further includes means for, if the receive
3 request is received from the device driver section, receiving the
4 receive request via a program interface with respect to the protocol
5 stack section.

1 15. A terminal device according to claim 5, wherein the
2 specific packet is a packet defined by a communication protocol

3 at higher than or equal to layer5 of OSI, which includes an RTP
4 packet.

1 16. A terminal device according to claim 3, wherein the
2 session information includes a MAC address corresponding to layer2
3 of OSI, a protocol number and an IP address corresponding to layer3
4 of OSI, and a port number corresponding to layer4 of OSI.

1 17. A terminal device according to claim 9, wherein the
2 first session information and the second session information
3 include a MAC address corresponding to layer2 of OSI, a protocol
4 number and an IP address corresponding to layer3 of OSI, and a
5 port number corresponding to layer4 of OSI.

1 18. A terminal device according to claim 10, wherein the
2 third session information includes a MAC address corresponding
3 to layer2 of OSI, a protocol number and an IP address corresponding
4 to layer3 of OSI, and a port number corresponding to layer4 of
5 OSI.

1 19. A terminal device according to claim 13, wherein the
2 program interface includes a NDIS interface.

1 20. A terminal device according to claim 13, wherein the
2 program interface includes a socket interface.

1 21. A method for processing communication data inside a
2 terminal device that includes: a device driver section for
3 controlling an interface section that is connected to a network;
4 and a protocol stack section that is connected to the network via

5 the device driver section based on a communication protocol at
6 higher than or equal to layer3 of OSI, the method comprising:
7 if a send request for a send frame is issued from the protocol
8 stack section, determining a send priority of the send request
9 based on header information at layer2 to layer4 of OSI within the
10 send frame; and
11 outputting the send request to the device driver section
12 according to the send priority determined.

1 22. A method for processing communication data inside a
2 terminal device according to claim 21, comprising:
3 if a receive request for a receive frame is issued from the
4 device driver section, determining a receive priority of the
5 receive request based on respective header information at layer2
6 to layer4 of OSI within the receive frame; and
7 outputting the receive request to the protocol stack section
8 according to the receive priority determined.

1 23. A method for processing communication data inside a
2 terminal device according to claim 21, further comprising:
3 if a send request for the send frame is issued from the protocol
4 stack section, determining a send priority of the send request
5 by searching a cache table in which high-priority session
6 information is previously registered based on respective header
7 information at layer2 to layer4 of OSI within the send frame;
8 queueing the send request to one of the multiple FIFOs each
9 corresponding to the send priority according to the send priority
10 determined; and
11 synthesizing send requests outputted from the multiple FIFOs
12 according to the send priority of the one of the multiple FIFOs

13 to which the send request is queued, and outputting a synthesized
14 send request to the device driver section.

1 24. A method for processing communication data inside a
2 terminal device according to claim 22, further comprising:

3 if a receive request for the receive frame is issued from
4 the device driver section, determining a receive priority of the
5 receive request by searching a cache table in which high-priority
6 session information is previously registered based on respective
7 header information at layer2 to layer4 of OSI within the receive
8 frame;

9 queueing the receive request to one of the multiple FIFOs
10 each corresponding to the receive priority according to the receive
11 priority determined; and

12 synthesizing receive requests outputted from the multiple
13 FIFOs according to the receive priority of the one of the multiple
14 FIFOs to which the receive request is queued, and outputting a
15 synthesized receive request to the protocol stack section.

1 25. A method for processing communication data inside a
2 terminal device that includes: a device driver section for
3 controlling an interface section that is connected to a network;
4 and a protocol stack section that is connected to the network via
5 the device driver section based on a communication protocol at
6 higher than or equal to layer3 of OSI, the method comprising:

7 if a send request to a predetermined destination for a
8 specific packet defined by a communication protocol at higher than
9 or equal to layer5 of OSI is issued from the protocol stack section
10 in advance and if the send request is a first one of consecutive
11 send requests, checking on header information of the specific

12 packet, then registering into a cache table session information
13 extracted from headers at layer2 to layer4 of OSI within a send
14 frame carrying the specific packet, raising a send priority of
15 the send request, and outputting the send request to the device
16 driver section; and

17 if the send request is among the consecutive send requests
18 other than the first one and if session information extracted from
19 headers at layer2 to layer4 of OSI within a send frame carrying
20 the specific packet is registered in the cache table, raising a
21 send priority of the send request, and outputting the send request
22 to the device driver section.

1 26. A method for processing communication data inside a
2 terminal device according to claim 25, further comprising:

3 if a receive request from a predetermined source for a
4 specific packet defined by a communication protocol at higher than
5 or equal to layer5 of OSI is issued from the device driver section
6 in advance and if the receive request is a first one of consecutive
7 receive requests, checking on header information of the specific
8 packet, then registering into a cache table session information
9 extracted from headers at layer2 to layer4 of OSI within a receive
10 frame carrying the specific packet, raising a receive priority
11 of the receive request, and outputting the receive request to the
12 protocol stack section; and

13 if the receive request is among the consecutive receive
14 requests other than the first one and if session information
15 extracted from headers at layer2 to layer4 of OSI within a receive
16 frame carrying the specific packet is registered in the cache table,
17 raising a receive priority of the receive request, and outputting
18 the receive request to the protocol stack section.

1 27. A method for processing communication data inside a
2 terminal device according to claim 25, further comprising:
3 if the session information is registered into the cache table,
4 monitoring the cache table; and
5 if the send request for a send frame carrying the session
6 information is not issued from the protocol stack section within
7 a predetermined time, deleting the session information within the
8 cache table.

1 28. A method for processing communication data inside a
2 terminal device according to claim 26, further comprising:
3 if the session information is registered into the cache table,
4 monitoring the cache table; and
5 if the receive request for a receive frame carrying the
6 session information is not issued from the device driver section
7 within a predetermined time, deleting the session information
8 within the cache table.

1 29. A method for processing communication data inside a
2 terminal device that includes: a device driver section for
3 controlling an interface section that is connected to a network;
4 and a protocol stack section that is connected to the network via
5 the device driver section based on a communication protocol at
6 higher than or equal to layer3 of OSI, the method comprising:
7 if a send request for a send frame is issued from the protocol
8 stack section, checking whether a first session information that
9 is extracted from headers at layer2 to layer4 of OSI within the
10 send frame is registered in a second cache table that is used upon
11 establishment of a session;

12 if the first session information is registered in the second
13 cache table, queueing the send request to a first FIFO section
14 for storing high-priority send data in a FIFO format;

15 if the first session information is not registered in the
16 second cache table, checking whether the first session information
17 is registered in a first cache table in which a second session
18 information is previously registered;

19 if the first session information is registered in the first
20 cache table, checking whether the send frame includes a
21 predetermined specific packet at higher than or equal to layer5
22 of OSI;

23 if the send frame includes the predetermined specific packet,
24 judging that a high-priority session is established, registering
25 the first session information into the second cache table, and
26 queueing the send request to the first FIFO section;

27 if the first session information is not registered in the
28 first cache table, queueing the send request to a second FIFO section
29 for storing low-priority send data in a FIFO format; and

30 outputting to the device driver section the send request
31 queued to the first FIFO section prior to the send request queued
32 to the second FIFO section.

1 30. A method for processing communication data inside a
2 terminal device according to claim 29, further comprising:

3 if a receive request for a receive frame is issued from the
4 device driver section, checking whether a third session information
5 that is extracted from headers at layer2 to layer4 of OSI within
6 the receive frame is registered in a second cache table;

7 if the third session information is registered in the second
8 cache table, queueing the receive request to a third FIFO section.

9 for storing high-priority receive data in a FIFO format;
10 if the third session information is not registered in the
11 second cache table, checking whether the third session information
12 is registered in a first cache table;
13 if the third session information is registered in the first
14 cache table, checking whether the receive frame includes a specific
15 packet;
16 if the receive frame includes the specific packet, judging
17 that a high-priority receive session is established, registering
18 the third session information into the second cache table, and
19 queueing the receive request to the third FIFO section;
20 if the third session information is not registered in the
21 first cache table, queueing the receive request to a fourth FIFO
22 section for storing low-priority receive data in a FIFO format;
23 and
24 outputting to the protocol stack section the receive request
25 queued to the third FIFO section prior to the receive request queued
26 to the fourth FIFO section.

1 31. A method for processing communication data inside a
2 terminal device according to claim 29, further comprising:
3 if the first session information is registered into the
4 second cache table, monitoring the second cache table; and
5 if the send request for a send frame carrying the first session
6 information is not issued from the protocol stack section within
7 a predetermined time, deleting the first session information within
8 the second cache table.

1 32. A method for processing communication data inside a
2 terminal device according to claim 30, further comprising:

3 if the third session information is registered into the
4 second cache table, monitoring the second cache table; and
5 if the receive request for a receive frame carrying the third
6 session information is not issued from the device driver section
7 within a predetermined time, deleting the third session information
8 within the second cache table.

1 33. A method for processing communication data inside a
2 terminal device according to claim 21, further comprising, if the
3 send request is issued to the device driver section, issuing the
4 send request via a program interface with respect to the protocol
5 stack section.

1 34. A method for processing communication data inside a
2 terminal device according to claim 22, further comprising, if the
3 receive request is received from the device driver section,
4 receiving the receive request via a program interface with respect
5 to the protocol stack section.

1 35. A method for processing communication data inside a
2 terminal device according to claim 25, wherein the specific packet
3 is a packet defined by a communication protocol at higher than
4 or equal to layer 5 of OSI, which includes an RTP packet.

1 36. A method for processing communication data inside a
2 terminal device according to claim 23, wherein the session
3 information includes a MAC address corresponding to layer 2 of OSI
4 within a frame, a protocol number and an IP address corresponding
5 to layer 3 of OSI, and a port number corresponding to layer 4 of
6 OSI.

1 37. A method for processing communication data inside a
2 terminal device according to claim 29, wherein the first and the
3 second session information includes a MAC address corresponding
4 to layer2 of OSI within a frame, a protocol number and an IP address
5 corresponding to layer3 of OSI, and a port number corresponding
6 to layer4 of OSI.

1 38. A method for processing communication data inside a
2 terminal device according to claim 30, wherein the third session
3 information includes a MAC address corresponding to layer2 of OSI
4 within a frame, a protocol number and an IP address corresponding
5 to layer3 of OSI, and a port number corresponding to layer4 of
6 OSI.

1 39. A method for processing communication data inside a
2 terminal device according to claim 33, wherein the program
3 interface includes a NDIS interface.

1 40. A method for processing communication data inside a
2 terminal device according to claim 33, wherein the program
3 interface includes a socket interface.

1 41. A program capable of being executed by a computer that
2 includes: a device driver section for controlling an interface
3 section that is connected to a network; and a protocol stack section
4 that is connected to the network via the device driver section
5 based on a communication protocol at higher than or equal to layer3
6 of OSI, the program comprising:

7 a process for, if a send request to a predetermined

8 destination for a specific packet defined by a communication
9 protocol at higher than or equal to layer5 of OSI is issued from
10 the protocol stack section in advance and if the send request is
11 a first one of consecutive send requests, checking on header
12 information of the specific packet, then registering into a cache
13 table session information extracted from headers at layer2 to
14 layer4 of OSI within a send frame carrying the specific packet,
15 raising a send priority of the send request, and outputting the
16 send request to the device driver section; and

17 a process for, if the send request is among the consecutive
18 send requests other than the first one and if session information
19 extracted from headers at layer2 to layer4 of OSI within a send
20 frame carrying the specific packet is registered in the cache table,
21 raising a send priority of the send request, and outputting the
22 send request to the device driver section.

1 42. A program according to claim 41 comprising:

2 a process for, if a receive request from a predetermined
3 source for a specific packet defined by a communication protocol
4 at higher than or equal to layer5 of OSI is issued from the device
5 driver section in advance and if the receive request is a first
6 one of consecutive receive requests, checking on header information
7 of the specific packet, then registering into a cache table session
8 information extracted from headers at layer2 to layer4 of OSI within
9 a receive frame carrying the specific packet, raising a receive
10 priority of the receive request, and outputting the receive request
11 to the protocol stack section; and

12 a process for, if the receive request is among the consecutive
13 receive requests other than the first one and if session information
14 extracted from headers at layer2 to layer4 of OSI within a receive

15 frame carrying the specific packet is registered in the cache table,
16 raising a receive priority of the receive request, and outputting
17 the receive request to the protocol stack section.

1 43. A program according to claim 41, comprising:
2 a process for, if the session information is registered into
3 the cache table, monitoring the cache table; and
4 a process for, if the send request for a send frame carrying
5 the session information is not issued from the protocol stack
6 section within a predetermined time, deleting the session
7 information within the cache table.

1 44. A program according to claim 42, comprising:
2 a process for, if the session information is registered into
3 the cache table, monitoring the cache table; and
4 a process for, if the receive request for a receive frame
5 carrying the session information is not issued from the device
6 driver section within a predetermined time, deleting the session
7 information within the cache table.

1 45. A program capable of being executed by a computer that
2 includes: a device driver section for controlling an interface
3 section that is connected to a network; a protocol stack section
4 that is connected to the network via the device driver section
5 based on a communication protocol at higher than or equal to layer 3
6 of OSI; a first cache table in which a first session information
7 is previously registered; a second cache table that is used upon
8 establishment of a session; a first FIFO section for storing
9 high-priority send data in a FIFO format; and a second FIFO section
10 for storing low-priority send data in a FIFO format, the program

11 comprising:

12 a process for, if a send request for a send frame is issued
13 from the protocol stack section, checking whether a second session
14 information that is extracted from respective headers at layer2
15 to layer4 of OSI within the send frame is registered in the second
16 cache table;

17 if the second session information is registered in the second
18 cache table, queueing the send request to the first FIFO section;

19 a process for, if the second session information is not
20 registered in the second cache table, checking whether the second
21 session information is registered in the first cache table;

22 a process for, if the second session information is
23 registered in the first cache table, checking whether the send
24 frame includes a predetermined specific packet at higher than or
25 equal to layer5 of OSI;

26 a process for, if the send frame includes the predetermined
27 specific packet, judging that a high-priority session is
28 established, registering the second session information into the
29 second cache table, and queueing the send request to the first
30 FIFO section;

31 a process for, if the second session information is not
32 registered in the first cache table, queueing the send request
33 to the second FIFO section; and

34 a process for outputting to the device driver section the
35 send request queued to the first FIFO section prior to the send
36 request queued to the second FIFO section.

1 46. A program according to claim 45, comprising:

2 a process for, if a receive request for a receive frame is
3 issued from the device driver section, checking whether a third

4 session information that is extracted from respective headers at
 5 layer2 to layer4 of OSI within the receive frame is registered
 6 in the second cache table;

7 if the third session information is registered in the second
 8 cache table, queueing the receive request to the third FIFO section
 9 for storing high-priority receive data in a FIFO format;

10 a process for, if the third session information is not
 11 registered in the second cache table, checking whether the third
 12 session information is registered in the first cache table;

13 a process for, if the third session information is registered
 14 in the first cache table, checking whether the receive frame
 15 includes a specific packet;

16 a process for, if the receive frame includes the
 17 predetermined specific packet, judging that a high-priority
 18 session is established, registering the third session information
 19 into the second cache table, and queueing the receive request to
 20 the third FIFO section;

21 a process for, if the third session information is not
 22 registered in the first cache table, queueing the receive request
 23 to the fourth FIFO section for storing low-priority send data in
 24 a FIFO format; and

25 a process for outputting to the protocol stack section the
 26 receive request queued to the third FIFO section prior to the receive
 27 request queued to the fourth FIFO section.

1 47. A program according to claim 45, comprising:

2 a process for, if the second session information is
 3 registered into the second cache table, monitoring the second cache
 4 table; and

5 a process for, if the send request for a send frame carrying

6 the second session information is not issued from the protocol
7 stack section within a predetermined time, deleting the second
8 session information within the second cache table.

1 48. A program according to claim 46, comprising:
2 a process for, if the third session information is registered
3 into the second cache table, monitoring the second cache table;
4 and
5 a process for, if the receive request for a receive frame
6 carrying the third session information is not issued from the device
7 driver section within a predetermined time, deleting the third
8 session information within the second cache table.

1 49. A program according to claim 41, comprising:
2 a process for, if the send request is sent to the device
3 driver section, outputting the send request via a program interface
4 with respect to the protocol stack section.

1 50. A program according to claim 42, comprising:
2 a process for, if the receive request is received from the
3 device driver section, receiving the receive request via a program
4 interface with respect to the protocol stack section.